

»Лекционен курс

»ООП1 (Java)



Избор: условни
оператори > 1



Въпроси, свързани с управление на програмите



Какво е контролен поток?

Последователността на изпълняваните в програмата оператори

Как можем да хореографираме контролния поток?

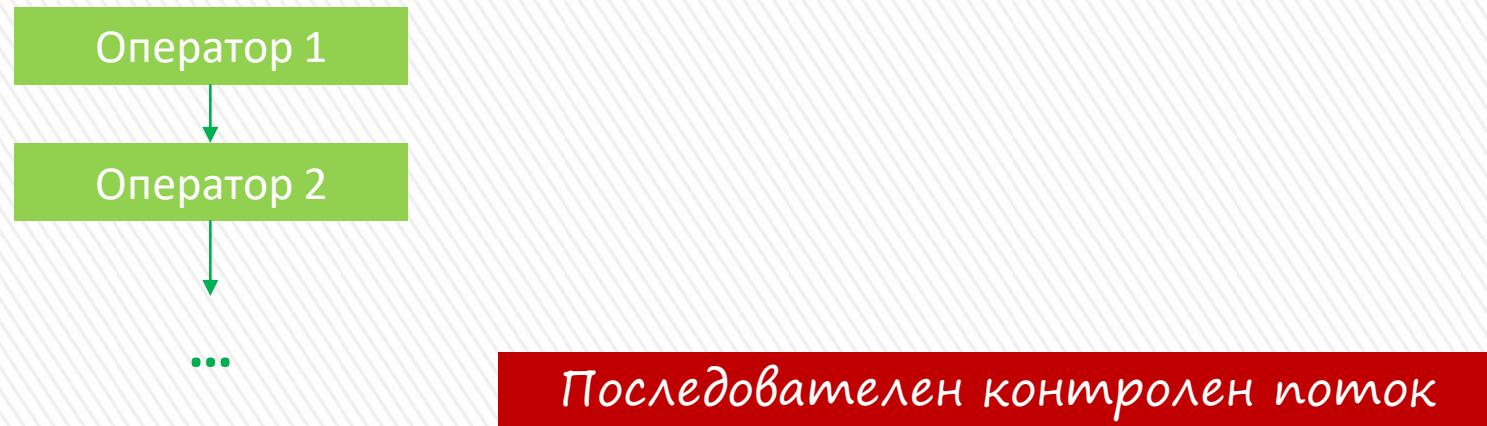
Оператори за:

- Цикли
- Условни оператори

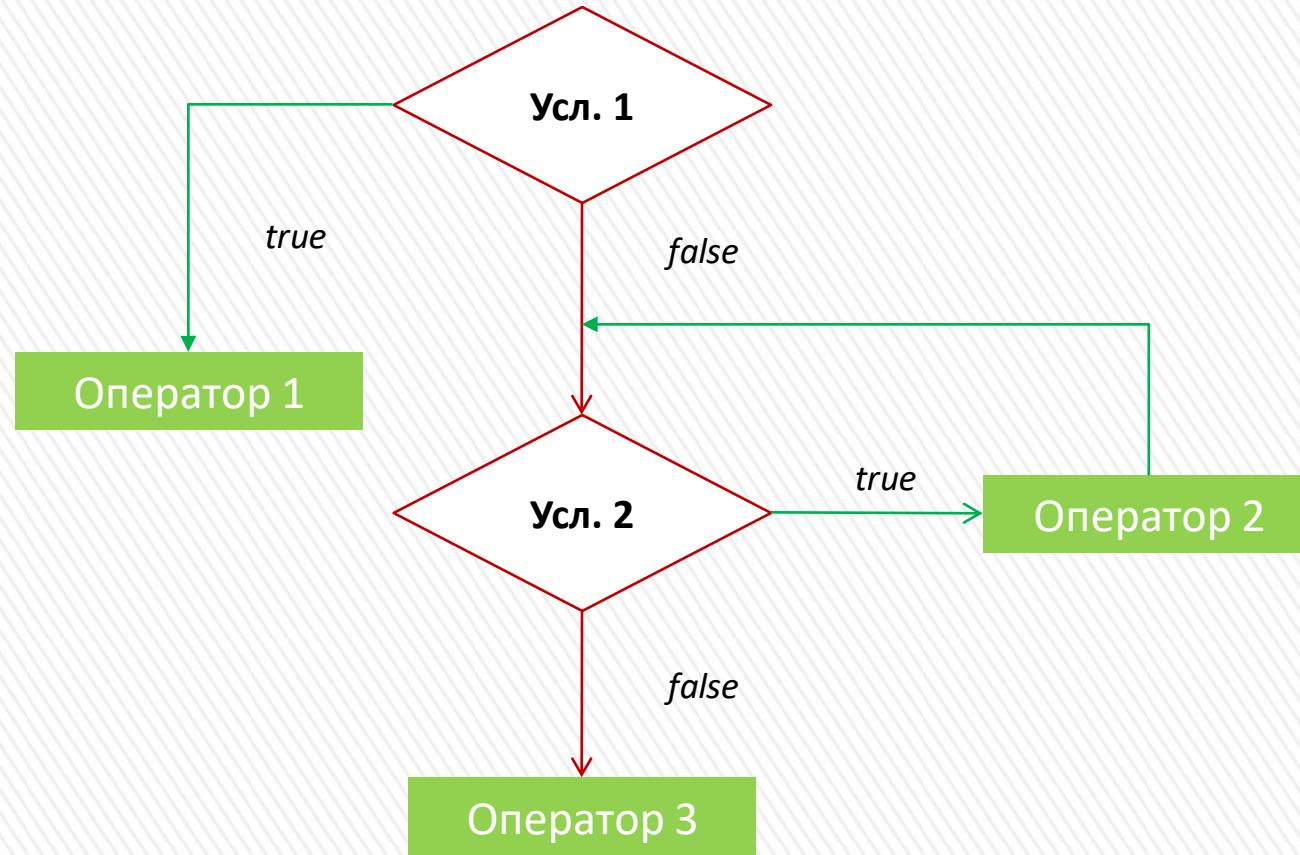
Контролен поток (control flow)



Какъв е контролният поток от примера?



Контролен поток с условия и цикли

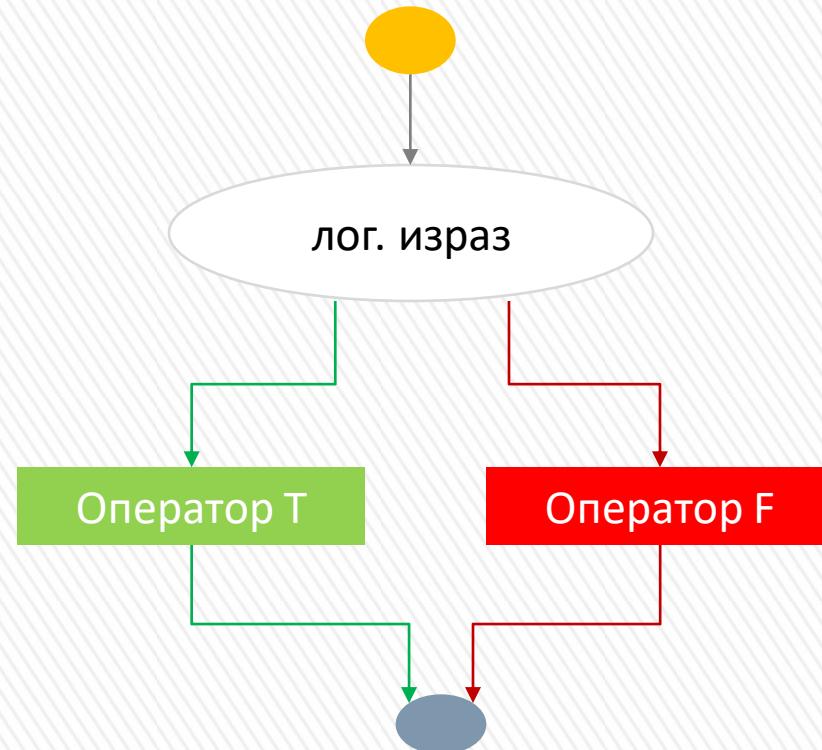


Управляващи конструкции

- » Процедурните езици за програмиране притежават някакъв вид управляващи конструкции
- » Често между отделните езици има препокриване
- » В Java ключовите думи за управляващи конструкции включват:
 - > if-else
 - > while
 - > do-while
 - > for
 - > switch

if оператор: разклоняваща структура

```
if (логически израз) {  
    оператор T;  
}  
else {  
    оператор F;  
}
```



Синтаксис

» If-оператор: избор между две алтернативи

EBNF: **if** (израз) оператор **else** оператор

За всяка алтернатива: един оператор !

Пример:

```
if (x == 0)
    System.out.print(0);
else
    System.out.print(y/x);
```

If-оператор: повече оператори

EBNF:

if (израз) оператор **else** оператор



Повече оператори?

с { . . . } обединени като един

```
if (x == 0)      { //Exception: Div by 0
    System.out.print(0);
    x = y;
}
else {
    System.out.print(y/x);
    y = x;
}
```

If-оператор: кратка форма

EBNF (пълна форма):

if (израз) оператор **else** оператор

```
if ((a + b) <= c)
    System.out.print(„не е триъгълник“);
else ; Празен оператор
```



Какво следва от синтаксиса?

EBNF (кратка форма):

if (израз) оператор

```
if ((a + b) <= c)
    System.out.print("не е триъгълник");
```

Тест: типични грешки



Какви грешки в примерите?

1

```
if a > b  
...  
else  
...
```

Липсват скоби

2

```
if (a = 1)  
...  
else  
...
```

Не е логическа стойност?

3

```
if (x > y)  
x = y; y = 0;  
if (y == 0)  
...
```

Стойност на y?

Пример



Какво прави програмата?

```
import java.util.Scanner;
public class BankBalance {
    public static final double OVERDRAWN_PENALTY = 8.00;
    public static final double INTEREST_RATE = 0.02; //2% annually
    public static void main(String[] args) {
        double balance;
        System.out.print("Въведете салдото по текущата си сметка:");
        Scanner keyboard = new Scanner(System.in);
        balance = keyboard.nextDouble();
        System.out.println("Оригинален баланс " + balance);
        if (balance >= 0)
            balance = balance + (INTEREST_RATE * balance)
                / 12;
        else
            balance = balance - OVERDRAWN_PENALTY;
        System.out.print("След корекция за един месец ");
        System.out.println("на лихви и неустойки, ");
        System.out.println("Вашият нов баланс е " + balance);
    }
}
```

Изчислява баланс на
сметки

Пример

- » В програмите, както и в ежедневието, понякога нещата могат да се развият по един от двата различни начина
- » Ако имаме пари в разплащащателна сметка, някои банки ще платят малка лихва
- » От друга страна, ако сме надвишили текущата си сметка и салдото по сметката е отрицателно и ще бъде начислена неустойка
- » Тази политика може да бъде отразена в счетоводната програма на банката чрез следния отчет на Java, известен като if-else извлечение

Пример



Какъв резултат?

```
import java.util.Scanner;
public class BankBalance {
    public static final double
    public static final double INTEREST_RATE = 0.02; //2% annually
    public static void main(Stri
        double balance;
        System.out.print("Въвед
        Scanner keyboard = new
        balance = keyboard.next
        System.out.println("Ори
        if (balance >= 0)
            balance = balance + (INTEREST_RATE * balance)
            / 12;
        else
            balance = balance - OVERDRAWN_PENALTY;
        System.out.print("След корекция за един месец ");
        System.out.println("на лихви и неустойки, ");
        System.out.println("Вашият нов баланс е " + balance);
    }
}
```

Въведете салдото по текущата си сметка:**120**
Оригинален баланс 120.0
След корекция за един месец на лихви и неустойки,
Вашият нов баланс е 120.2

Process finished with exit code 0

Въведете салдото по текущата си сметка:**-120**
Оригинален баланс -120.0
След корекция за един месец на лихви и неустойки,
Вашият нов баланс е -128.0

Process finished with exit code 0

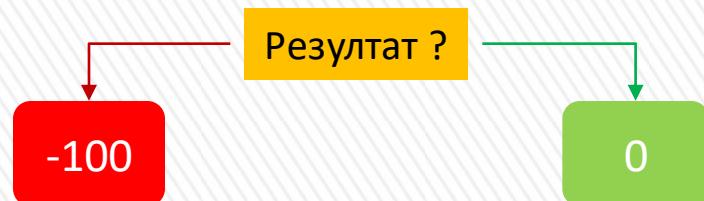
“Висяще” else (dangling else)

Входни стойности:

x	y	z	ok
2	1	100	-100

```
if (x > y)
    if (y > z)
        ok = 1;
else
    ok = 0;
```

```
if (x > y)
    if (y > z)
        ok = 1;
else
    ok = 0;
```



“Висящ” else: проблем



Разлики между двете версии?

1

```
if (a)  
    if (b)  
        s1;  
  
else  
    s2;
```

2

```
if (a)  
    if (b)  
        s1;  
  
else  
    s2;
```

Двета варианта са идентични

Двата варианта: идентични

Само по различен начин изразен (Layout)

→ намерение на програмиста:

else s2; принадлежи към ...

- 1. Форма: външен if

- 2. Форма: вътрешен if

Значение на една програма: независимо от Layout



Какво правим?

```
if (a)      if (b)    s1;    else s2;
```

2-та форма е обвързваща !

“Висящ”-else-проблем: причина

Граматиката на Java не е еднозначна !

Оператор за избор ::=

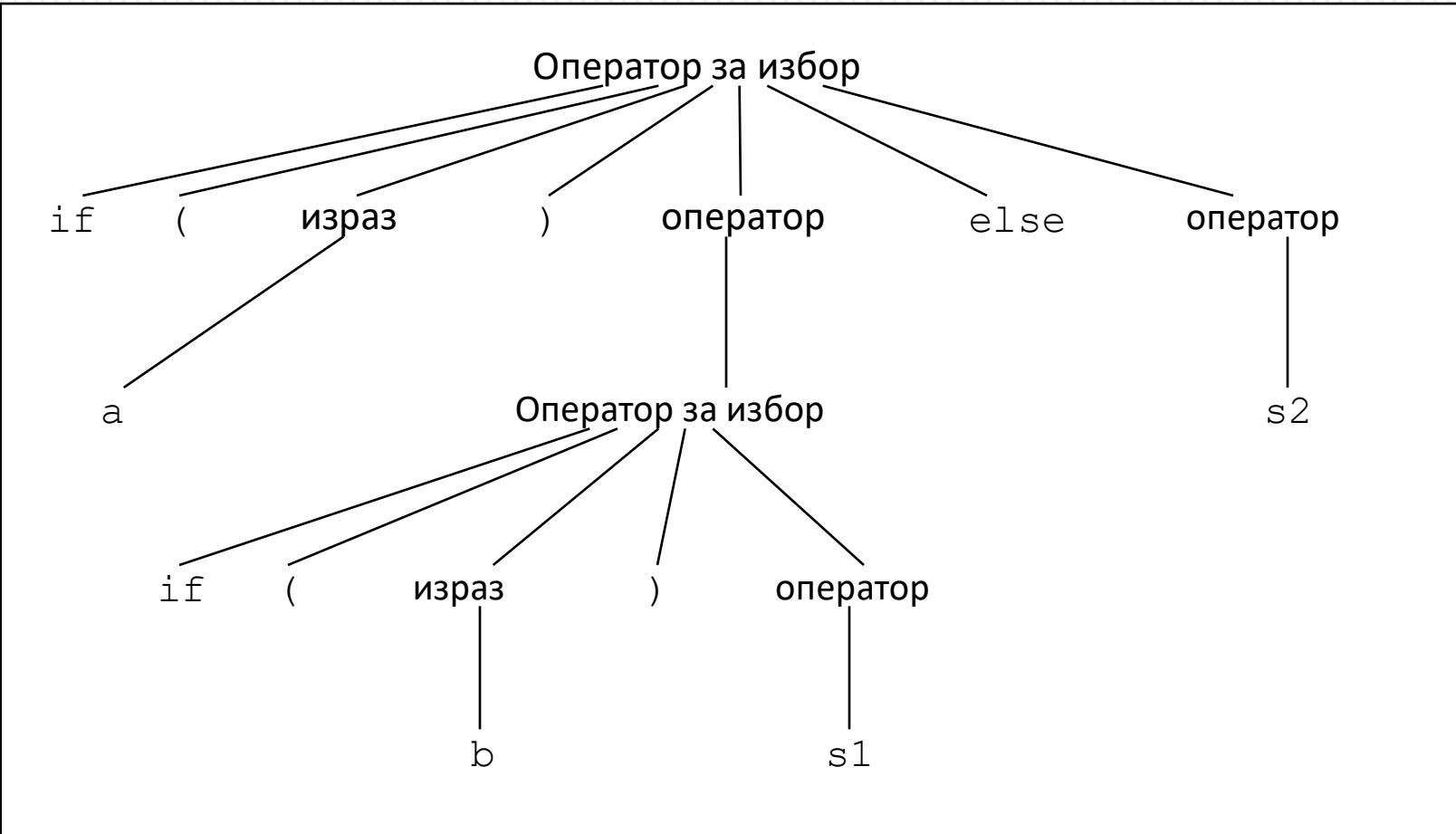
```
if ( израз ) оператор [ else оператор ] |  
switch ...
```

3 правила

Оператор за избор ::=

```
if ( израз ) оператор |  
if ( израз ) оператор else оператор |  
switch ...
```

Синтактично дърво: Вариант 1



Изведен:

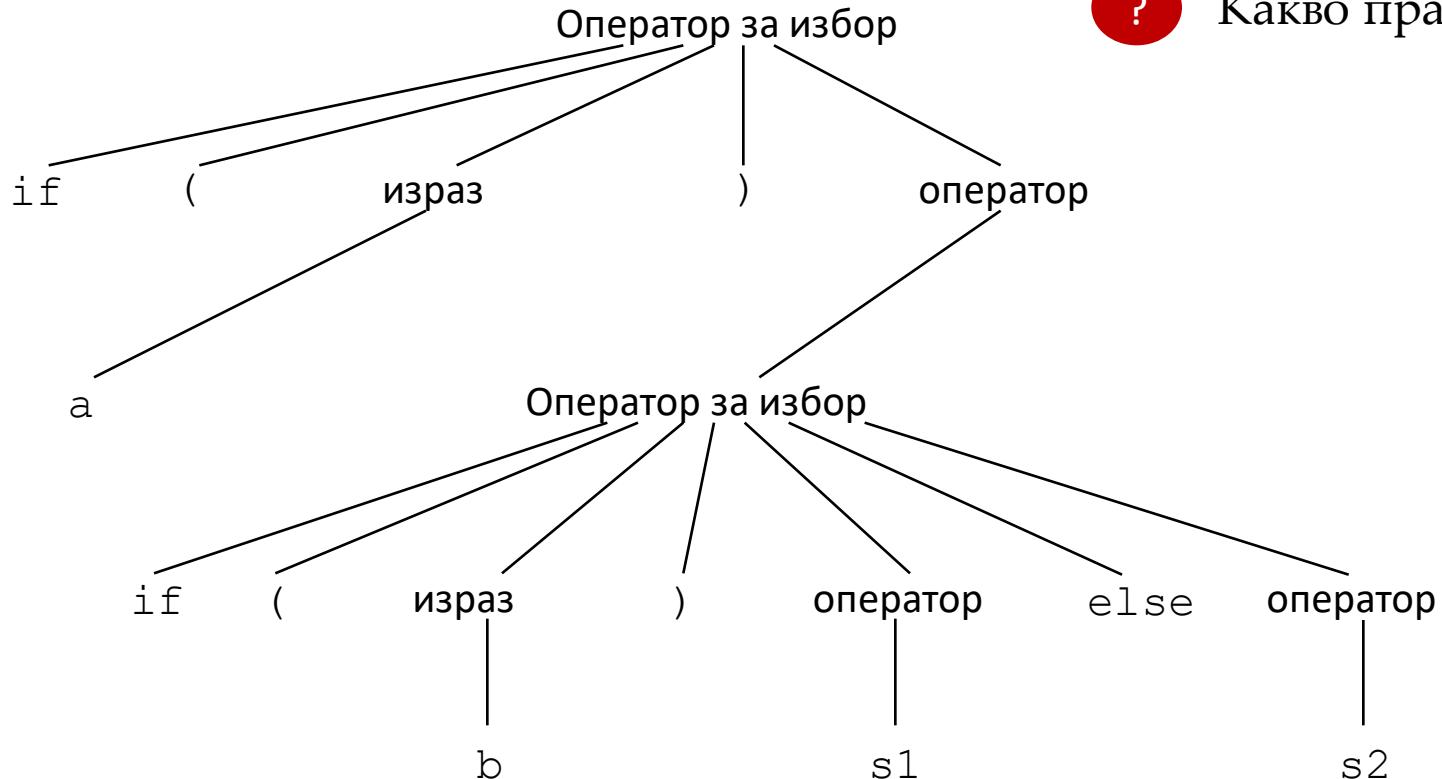
```
if ( a ) if ( b ) s1 else s2
```

Синтаксично дърво: Вариант 2

Две синтактични дървета за същия оператор



Какво правим?



Изведен:

if (a) if (b) s1 else s2

Проблеми

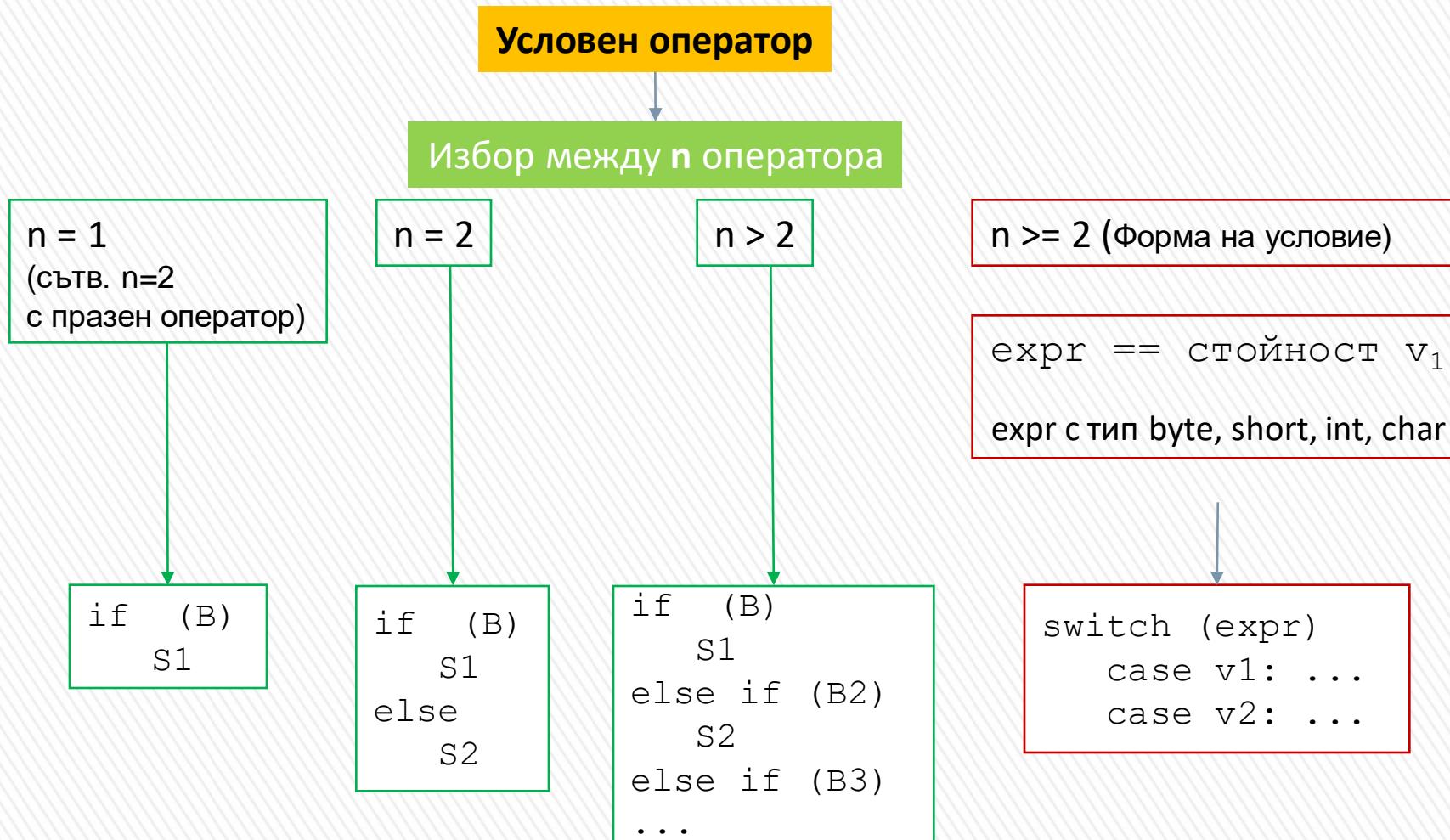


Какво правим, ако все пак искаме вариант 1 ?

```
if (x > y) {  
    if (y > z)  
        ok = 1; }  
else  
    ok = 0;
```

```
x > y)  
(y >  
ok =  
if (x > y)  
    if (y > z)  
        ok = 1;  
else  
= 0;  
else  
    ok = 0;
```

Избор на условен оператор: switch



Switch-оператор: мотивация

```
if (n == 0)
    System.out.print(" нула ");
else if (n == 1) 
    System.out.print(" едно ");
else if (n == 2)
    System.out.print("две");

. . . // else if до 9

else // n > 9
    System.out.print(" > девет ");
```

Стойност на израз
(променлива n)

Повтарящо тестване при
равенство с
предварително зададени
стойности (0, 1, 2 ...)

Switch-оператор: избор от повече варианти

```
switch (n) {  
    case 0: System.out.print( "0" );  
              break;  
    case 1: System.out.print( "1" );  
              break;  
    case 2: System.out.print( "2" );  
              break;  
    ....  
    default: System.out.println( " >9" );  
}
```

➔ Семантиката се запазва

Switch: пример



Коментар на примера?

```
final int jan = 1, feb = 2, mar = 3, ... dec = 12;  
int month, year, numDays;  
... // read month, year  
switch (month) {  
    case feb:  
        if (((year % 4) == 0) && ((year % 100) != 0)  
            || ((year % 400) == 0)) //high year test  
            numDays = 29;  
        else numDays = 28;  
        break;  
    case apr: case jun: case sep: case nov:  
        numDays = 30; break;  
    default: numDays = 31;  
}
```

Пример



Какъв резултат?

```
int n = 1;  
switch (n) {  
    case 0: System.out.print( "0" );  
    case 1: System.out.print( "1" );  
    case 2: System.out.print( "2" );  
    case 3: System.out.print( "3" );  
}  
System.out.println();
```

123

Break-оператор

- » Предназначение на break
 - > Завършва изпълнението на оператора switch в разклонението, където е даден
- » По принцип всяка алтернатива на switch трябва да има break
 - > При пропускане изпълнението на switch продължава до неговия край

Break-оператор



Какъв резултат?

```
int n = 1;  
switch (n) {  
    case 0: System.out.print( "0" ); break;  
    case 1: System.out.print( "1" ); break;  
    case 2: System.out.print( "2" ); break;  
    case 3: System.out.print( "3" ); break;  
}  
System.out.println();
```

1

Обобщение

Оператор за избор ::=

```
if ( израз ) оператор |  
if ( израз ) оператор else оператор |  
switch ...
```

Сравняване на низове



Какво прави програмата?

```
import java.util.Scanner;
public class StringEqualityDemo {
    public static void main(String[] args) {
        String s1, s2;
        System.out.println("Въведете два реда текст:");
        Scanner keyboard = new Scanner(System.in);
        s1 = keyboard.nextLine();
        s2 = keyboard.nextLine();
        if (s1.equals(s2))
            System.out.println("Двата текста са еднакви.");
        else
            System.out.println("Двата текста не са еднакви.");
        if (s2.equals(s1))
            System.out.println("Двата текста са еднакви.");
        else
            System.out.println("Двата текста не са еднакви.");
        if (s1.equalsIgnoreCase(s2))
            System.out.println("Текстовете са еднакви, игнорирайки
                                малки и големи букви.");
        else
            System.out.println("Текстовете не са еднакви, дори
                                игнорирайки малки и големи букви.");
    }
}
```

Сравнява символни
низове

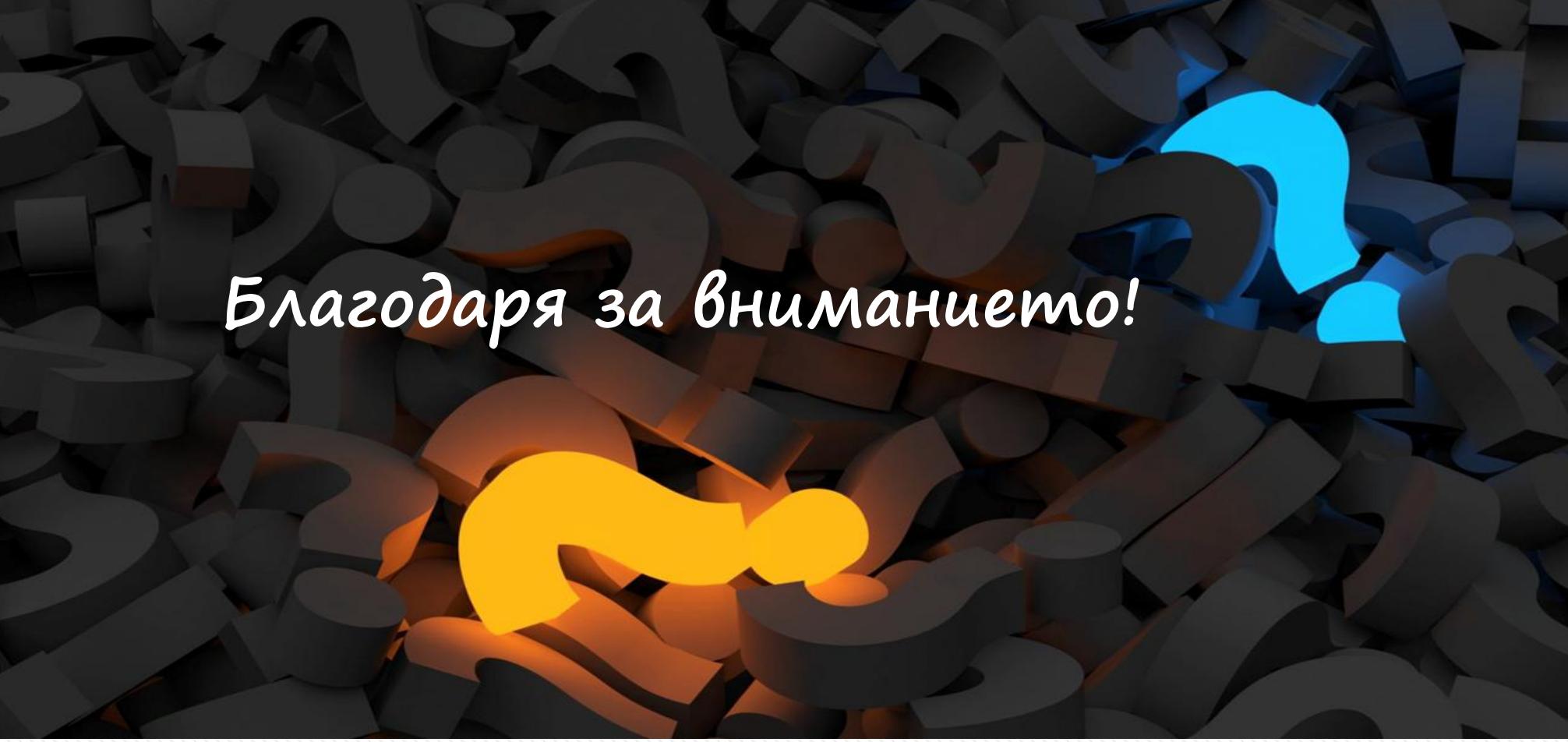
Сравняване низове



Какъв резултат?

```
import java.util.Scanner;
public class StringEqual {
    public static void main(String[] args) {
        String s1, s2;
        System.out.print("Въведете два реда текст:");
        Scanner keyboard = new Scanner(System.in);
        s1 = keyboard.nextLine();
        s2 = keyboard.nextLine();
        if (s1.equals(s2))
            System.out.println("Двета текста са еднакви.");
        else
            System.out.println("Двета текста не са еднакви.");
        if (s2.equals(s1))
            System.out.println("Двета текста са еднакви.");
        else
            System.out.println("Двета текста не са еднакви.");
        if (s1.equalsIgnoreCase(s2))
            System.out.println("Текстовете са еднакви, игнорирайки
                                малки и големи букви.");
        else
            System.out.println("Текстовете не са еднакви, дори
                                игнорирайки малки и големи букви.");
    }
}
```

Въведете два реда текст:
Аз съм в Пловдив.
Аз съм в София.
Двета текста не са еднакви.
Двета текста не са еднакви.
Текстовете не са еднакви, дори игнорирайки малки и големи букви.
Process finished with exit code 0



Благодаря за внимание!